TO: MSPM Distribution
FROM: Edwin W. Meyer, Jr.
SUBJECT: BE.18.00, BE.18.01
DATE: 09/23/68

The attached re-issue of BE.18.00 and the accompanying
BE.18.01 replace the Abstracts given in BE.18.00 of 07/02/68
and BE.18.00A of 07/24/68. A new implementation of the
merge_edit command is described, but its usage is
fundamentally unchanged.

BE.18.00 describes the use of the Multics merge_edit command
and the format of its input control file. Familiarity
with the CTSS MRGEDT command, described in BE.5.02, is
assumed.

BE.18.01 is a brief description of the merge_edit procedures
and their calling sequences.

Identification

The merge_edit Command
Edwin W. Meyer, Jr.

Purpose

In a manner similar to the CTSS mrgedt command, merge_edit
directs the creation of an IMCV tape on Multics which
can be run under GECOS to perform compilations, assemblies,
etc., and produce a tape by which these results can be
returned to Multics.

Description

The merge_edit command operates in two passes. During
pass 1 it scans a control file to produce an internal
list structure and other data. An ascii and a binary
control segment are produced during pass 2 from the data
structure and placed in the current working directory.
The tape daemon is signalled to create the IMCV tape,
and merge_edit returns to command level. It does not
produce the tape by itself or in the same process.

Usage

To execute a merge_edit, the user types

        merge_edit g_name runname username -opt1- -opt2-

where the arguments are interpreted as follows:

g_name   indicates gecos file to use in merge_edit. Either
            full segment name or only primary component is
            acceptable. The second component of the name of
            the gecos control segment must be ".gecos". g_name
            may be a pathname. If not, the segment is assumed
            to reside in the working directory.

runname   a 1 to 6 character string used to identify the
            job. This forms the primary component of the
            names of the two control segments that are produced
            in the working directory:

            runname.control and runname.control.binary

username   a 1 to 12 character string which identifies the user.

opt1, opt2   optional arguments. The following are currently defined:

    1. "mac" - means run this job at MAC

    2. "mh" - means run this job at Murray Hill

    3. "notape" - means tape daemon not to be signalled

    The options can appear in any order.

Both "mrgedt" and "mg" are acceptable abbreviations for "merge_edit".

To notify the tape daemon to execute the control segments produced by a previous merge_edit, the user types

    merge_edit runname (tape)

where runname is the runname of the previous merge_edit, and "(tape)" is a literal.

The Gecos Control File

Although it is generally quite similar to that described in BE.5.02 the gecos control file for the Multics version has several notable global differences:

    a.   The file is typed in lower case ascii, and an underscore (_) in a segment name must not be represented as a dash (-).

    b.   The merge_editor uses actual segment pathnames instead of 6 character filenames. However, since GECOS requires six character file names, the merge_editor converts the entry names of such pathnames into guaranteed unique filenames. If the pathname is six characters or less no conversion is done.

    c.   The compilation control lines bcpl, epl, and tmgl accept a maximum of four options.

There are several new or altered control lines:

1.   bcpl (abbrev: bc) and tmgl are used in the same
      manner as the epl control line to direct bcpl and
      tmgl compilations.

2.   comment is not currently implemented.

3.   deck name_1 name_2 ... name_j

      Any number of segments may be decked in one line.

4.   fetch name_1 class_1 name_2 class_2 ... name_j
      -classj-

      The class tokens may be either "*" (text, link,
      symbol, and list segments are returned), or "tl"
      (text, link, and symbol segments returned).
      If the final class token on the line is absent the
      effect is as though it were "tl".

5.   insert alpha

      alpha.gecos is inserted.

6.   maketl segname -opts- / text+link segname -opts-

      segname has the general format

           path>name:p_name

      This specifies that the segment "name" (found in
      the working directory if the optional component
      -path>- is absent) is to be loaded as "p_name" in the
      pseudo-process. However, if the final component
      -:p_name- is absent, the segment is loaded as
      "name".

      The text+link control line handler expects link
      and symbol segments to be named name.link and
      name.symbol respectively.

      maketl *:name -opts-

      directs the creation of a dummy (empty) text and
      link segment to be loaded as name.

Use the pseudo-process segment name ("p_name"
if present - otherwise "name") in deck/fetch
control lines for text+link-ed or maketl-ed
segments.

7.   mst -mste_name-

creates a control card for an mst tape, and if
the optional 1 to 12 character -mste_name- is
present, also creates a control card for an mste
tape of that name. This line is for the use of
the mst generator and editor.

8.   No text+link-ed symbol segments will be loaded unless
the control line

        symbol

is present. In the future this may also apply to
libe-ed segments.

9.   The control line

        undump

has no effect. The undump switch is on by default,
and no pseudo-process dump is ever put on the return
tape.

Example

```
epl   fix_mg_tlmk
eplbsa system_archive>mg_tables
tl   fix_merge_edit:merge_edit   slvacc slvprc
tl   >system_library>mg_comp
tl   mg_ldlb
li   mg_pass1
load * slvprc   slvacc
fetch *
deck * merge_edit mg_comp mg_ldlb mg_pass1
```